SPAWAR

19th Annual ICCRTS

C2 Agility: Lessons Learned from Research and Operations

Primary Topic: Data, Information, and Knowledge
Alternate Topic: Experimentation, Metrics, and Analysis
Alternate Topic: Social Media

Printed February 7, 2014

Empirical determination of pattern match confidence in labeled graphs

Dr. Ben Migliori, Dr. Daniel Grady, and Dr. James Law

Prepared by
Space and Naval Warfare Systems Center, 53560 Hull Street, San Diego, CA 92152-5001, USA
Point of Contact: Dr. Ben Migliori. benjamin.migliori@navy.mil 619-553-3269

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
07 FEB 2014		**00-00-2014 to 00-00-2014**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Empirical determination of pattern match confidence in labeled graphs	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Space and Naval Warfare Systems Center,53560 Hull Street,San Diego,CA,92152-5001**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

Presented at the 18th International Command & Control Research & Technology Symposium (ICCRTS) held 16-19 June, 2014 in Alexandria, VA.

14. ABSTRACT

The ability to represent complex, arbitrary situations in terms of labeled graphs has profound implications for situational awareness across domains. However, such graphs are fundamentally difficult for manual processing by experts; although our visual system typically outperforms all algorithms for pattern detection a graph with a few as several hundred nodes and edges reveals very little upon visual inspection. Thus we are forced to rely on pattern- matching algorithms to extract meaning from graph representations where nodes, labels and edges represent specific entities, general categories, and relationships. Algorithms such as Complex Event Processing (CEP), search a graph for a particular set of relationships between the categories that make up the ontology of the situation. In this paper, we will present an empirical method for determining the likelihood that a pattern search will return a false positive for a given pattern, ontology, and graph. This likelihood is analogous to the signalto- noise ratio in traditional sensing schemes. We demonstrate our method using algorithmically generated datasets and in datasets with known ground truths. We also show scale-free (power-law) behavior in several graph types, which allows for estimation of maximum graph size before false positives are expected to occur. Finally, we present a preliminary analytical study that describes the number of arbitrary pattern matches expected to appear by chance in a larger labeled graph. In any operationally relevant situation, assigning a confidence or quality metric to data used for decision making is crucial. The method presented in this paper is one of the first methods for doing so with complex patterns detected in large, highly-interrelated datasets. We believe that an improved understanding of pattern match quality will improve the usefulness of search techniques applied to social media, operation intelligence and tactical intelligence, while helping to encourage the adoption of modern analysis techniques for non-human-readable information.

15. SUBJECT TERMS					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**15**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

19th Annual ICCRTS
C2 Agility: Lessons Learned from Research and Operations
Primary Topic: Data, Information, and Knowledge
Alternate Topic: Experimentation, Metrics, and Analysis
Alternate Topic: Social Media
Printed February 7, 2014

Empirical determination of pattern match confidence in labeled graphs

Dr. Ben Migliori, Dr. Daniel Grady, and Dr. James Law
Space and Naval Warfare Systems Center
53560 Hull Street
San Diego, CA 92152-5001, USA
619-553-2369
benjamin.migliori@navy.mil

Abstract

The ability to represent complex, arbitrary situations in terms of labeled graphs has profound implications for situational awareness across domains. However, such graphs are fundamentally difficult for manual processing by experts; although our visual system typically outperforms all algorithms for pattern detection, a graph with a few as several hundred nodes and edges reveals very little upon visual inspection. Thus, we are forced to rely on pattern- matching algorithms to extract meaning from graph representations where nodes, labels and edges represent specific entities, general categories, and relationships. Algorithms, such as Complex Event Processing (CEP), search a graph for a particular set of relationships between the categories that make up the ontology of the situation.

In this paper, we will present an empirical method for determining the likelihood that a pattern search will return a false positive for a given pattern, ontology, and graph. This likelihood is analogous to the signal-to-noise ratio in traditional sensing schemes. We demonstrate our method using algorithmically generated datasets and in datasets with known ground truths. We also show scale-free (power-law) behavior in several graph types, which allows for estimation of maximum graph size before false positives are expected to occur. Finally, we present a preliminary analytical study that describes the number of arbitrary pattern matches expected to appear by chance in a larger labeled graph.

In any operationally relevant situation, assigning a confidence or quality metric to data used for decision making is crucial. The method presented in this paper is one of the first methods for doing so with complex patterns detected in large, highly-interrelated datasets. We believe that an improved understanding of pattern match quality will improve the usefulness of search techniques applied to social media, operation intelligence, and tactical intelligence, while helping to encourage the adoption of modern analysis techniques for non-human-readable information.

Introduction

Situational awareness and understanding presents one of the most important challenges for the modern military. Even as recently as World War II [1], the amount of intelligence information acquired via all available streams was small enough that individuals and small teams were able to process, understand, and synthesize that information to make informed decisions. This process used peer review (although not always effectively [10] [12]) as a method of establishing the quality of both the intelligence and the recommendations [15]. However, this process has become substantially more difficult as the military has shifted to network-centric warfare [11] [3] [8]. The number of sensors, the number of platforms, and the speed of information transmission have all substantially increased; this results in a tremendous mass of continuously acquired measurements that must be converted into a situational description, and then analyzed to create situational understanding. The amount of information overwhelms the capacity of individual trained analysts, and continues to grow. Worse, attempting to parallelize the work among many analysts may not solve the problem; by dividing a set of interconnected items among many individuals, the critical links that lead to a full understanding may be arbitrarily severed. The problem becomes a form of Catch-22: we need to analyze intelligence to identify important links, but to analyze the intelligence, we need to know which links are important.

A possible solution to this paradox is to analyze the entire set of information simultaneously [9] [13] [17] [14] [16]. A full description of an arbitrary situation can be represented by a mathematical structure called a labeled graph, in which the relationships between arbitrary entities (represented as nodes) are indicated by pairwise links. In a labeled graph, a known event of interest will appear as a specific structure (i.e. a specific set of labeled nodes and edges). By defining an ontology (a hierarchical set of categories) to which the labels belong, an unknown event of interest can be identified by matching an event pattern to a specific set of nodes and edges in the graph. These event patterns are defined by structural relationships between the elements of the ontology (the categories of the labels). Thus, a pattern describes a class of event that is to be identified, without naming the specific entities that might make up that event, and a pattern match occurs when the specific structure and categories in the data graph match the general pattern. [5]

An example describing a hypothetical event pattern is shown in Figure 1A with a diagram of a corresponding ontology. This pattern would be an extremely generic example, in which any incoming ship carrying anything that could be construed as a threat would trigger a warning on arrival to a military port. A data graph with a set of unique nodes and edges whose categorical types match the pattern (Fig. 1A) is shown in Figure 1B. However, even in this simple example, the analysis can become complicated. Figure 1C shows a similar data graph for which there are no subsets of graph node and links matching the pattern exactly. We can extend the search by defining a range of neighbors in which to search; for example, two nodes might be defined as a match if they are neighbors-of-neighbors and belong to the correct category, even if the interstitial nodes are not part of the pattern. In Figure 1C, a pattern match is found by allowing one interstitial node in the result, thus creating a virtual edge shown as a dashed red line. This increases the sensitivity to events of interest, such as the arrival of a suspect at a particular port coinciding with the departure of a ship at that port as shown in the example.

Although these examples can be quickly interpreted, large graphs (more than $\approx$ 50 - 200 nodes, Figure 1D) become fundamentally difficult for the human eye to process. These hairballs reveal

very little on visual inspection, and require computational tools to understand. We are thus forced to rely on pattern-matching algorithms to extract meaning in the form of specific subgraphs matching a general search pattern. This is not a critical flaw; the ability to represent complex, arbitrary situations in terms of labeled graphs has profound implications specifically because it is amenable to computational analysis. After a pattern is successfully identified, a new problem arises. Within a given labeled graph, with labels drawn from a specific set of categories, what is the likelihood that a search pattern will result in a successful match by chance? This problem becomes extremely difficult as the graph size increases. Conceptually, the false positive likelihood is related to the level of connectivity of the graph, the size of the graph, the number of label categories, and the number of allowed interstitial nodes in the pattern. False positives may arise from a pattern that is too simple, a graph that is too small, or a category set which does not successfully resolve the entities. It is also entirely possible that in a situation where many pattern matches do exist, they could be interpreted as false positives without additional analysis or comparison to guide the user.

In this paper, we investigate the dependence of subgraph isomorphism count (the number of pattern matches) as a function of ontology and graph size for several canonical graph types. Our results show scale-free (power-law) behavior, which may in certain cases may allow for estimation of maximum graph size before false positives are expected to occur. Our research suggests an empirical method for determining the likelihood (analogous to signal to noise ratio) that a pattern search will return a false positive for a given pattern, ontology, and graph structure. This empirical method may prove useful in assigning confidence boundaries to pattern search results, providing a valuable tactical decision aid in the context of modern situational awareness.

Methods

Algorithmic generation of graphs and ontologies. Graphs were generated using the open source graph analysis software Gephi 0.9 (https://gephi.org/about/). The plugin program Complex Graph Generators (https://gephi.org/plugins/complex-generators/) was used to generate the specific graph structures. Three graph types were explored; Erdős–Rényi [6] random graphs, Barabási–Albert preferential attachment graphs [2], and Watts–Strogatz [18] small world graphs. The ER graphs were given a fixed edge probability of 0.05. The parameters for the Barabási–Albert were one edge added or rewired per step, and a 0.45 probability of adding or rewiring. The parameters for the SW graphs were alpha = 0.05, average degree 4. Each graph was generated in an unlabeled state with 50, 500, 1000, and 2500 nodes. Each node and edge was assigned a unique two-letter name drawn in sequential order.

A series of ontologies were then produced in the form of branching tree structures generated by the Gephi program. The sizes and label distributions of the ontological trees are shown in Table 1. Each ontology represents a different level of *expressiveness* or *resolution*; the more unique categories and divisions between categories, the more accurate a situational description will be.

Surrogate data generation. To assign categorical meaning to the labeled graphs, categories were assigned to nodes and edges in the graph from the ontologies according to a uniform distribution, using a custom Java program based on the Gephi API. For each graph size and ontology

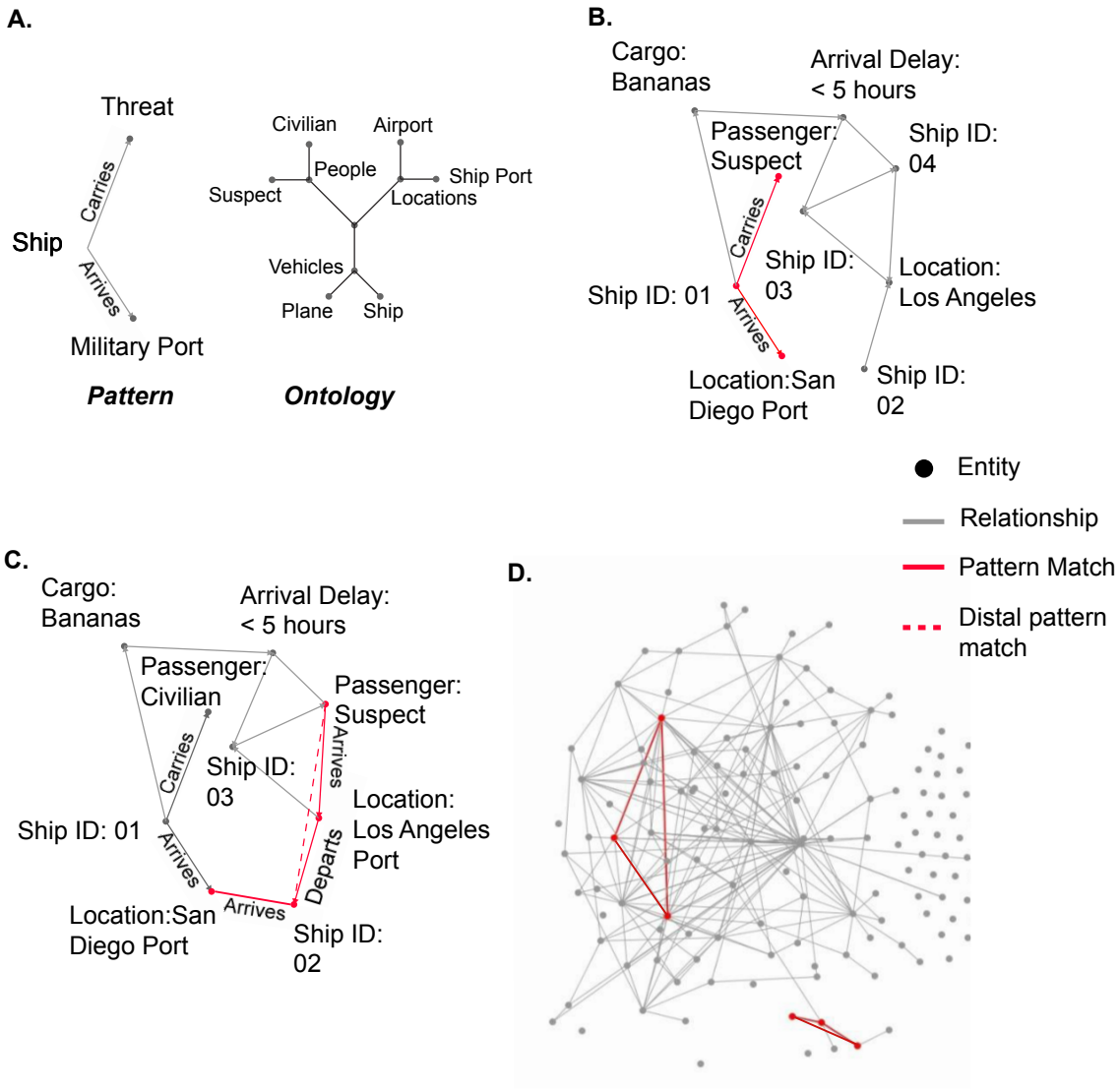

Figure 1. Pattern search of artificial labeled graphs. (A) An example of a pattern suitable for searching a graph for a complex event. This pattern is contructed from the elements in an ontology, a simplified form of which is shown at right. (B) An example of a hypothetical data graph containing the event of interest described in (A). The event is present as an isomorphic subgraph and is highlighted in red. The other nodes and edges are confounding information in the search that must be discarded. (C) An example of a hypothetical data graph containing a non-isomorphic event of interest. In this case, there is an interstitial node between elements that match the pattern. When pattern search is performed allowing matches within 2 nodes, the pattern highlighted in red is found. The dashed line represents the distal pattern match, and indicates a virtual edge that would appear in the result. (D) A larger dataset with pattern matches highlighted. Datasets of this size or larger are very difficult to label and describe.

Resolution	Nodes	Branches
Minimal	16	13
Low	32	28
High	64	28
Ultra	256	56

Table 1. Ontology resolution. Each ontology had a specific number of described categories, defining an effective resolution. Note that only the minimal and low resolution ontologies, shown in Fig. 2, produced enough pattern matches for further analysis.

resolution, this process was repeated four times. This created four trials for each configuration on which the pattern matching algorithm could be run.

A search pattern was created in the form of an acyclic triangular graph. For each ontology shown in Table 1, this graph was relabeled with categories drawn from a uniform distribution in the same manner as for the data graphs. Thus for each set of categories, the search pattern was both random and unique. This method represents the creation of surrogate graph data retaining the same statistical properties as the original data.

Pattern search algorithm. To perform pattern searches of our labelled graphs, we used the method of Complex Event Processing, an implementation of the method described by [7]. Details about the algorithm may be found in [9]. Briefly, the algorithm can be written in pseudocode as

Inpute: Pattern graph $P = (V_p, E_p)$, Data graph $G = (V, E)$, and ontology O.

Initialize by determining shortest path between all pairs.

1. Find descendent nodes of each pattern graph node, $D_{vp} = desc(V_p)$, in the ontology O.
2. Compute potential matches M_{vp} in G for each node in D_{vp}.
3. Transverse paths in M_{vp}, while storing path length and edge type.
4. Remove nodes that are not connected.
5. Remove nodes that do not meet path length (interstitial node distance) constraints.
6. Remove nodes that do not have the correct edge type in the ontology O.

At the conclusion of the algorithm, the search results contains all nodes and edges that match the pattern. This search result does not separate individual subgraph isomorphisms, but rather produces at least one graph which may be composed of several linked isomorphisms. To determine the specific number of fully matching subgraphs, a brute force subgraph isomorphism separation algorithm (http://networkx.github.io/index.html, [4]) was run on the pattern match result.

Results

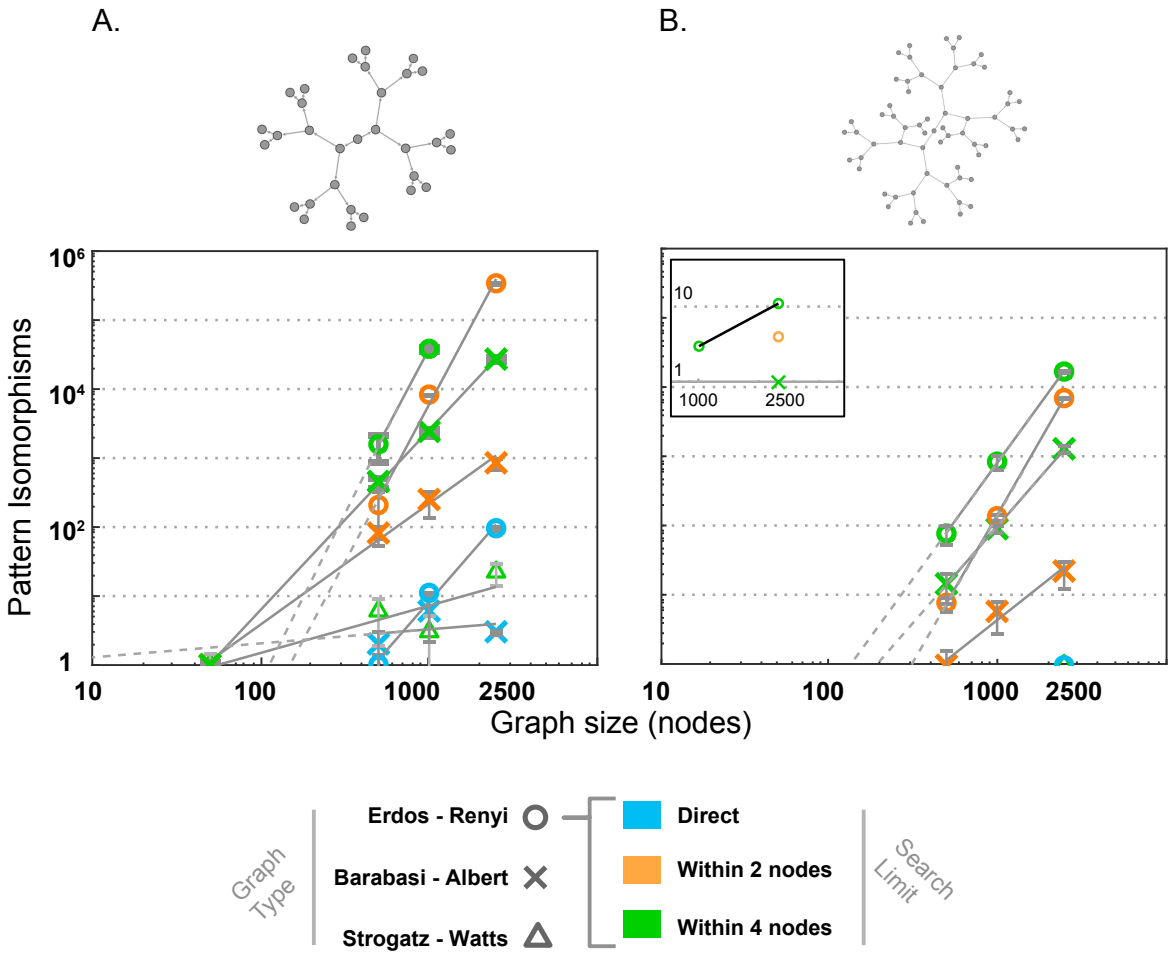

Figure 2. Results of pattern search on algorithmically generated random labeled graphs. Erdős–Rényi (ER, circles), Barabási–Albert (BA, crosses) and Watts–Strogatz (WS, triangles) graphs were generated with sizes ranging from 50 to 2500 nodes, and labeled algorithmically with an ontology. For each graph type and ontology, pattern search was performed using the CEP algorithm with the interstitial node limit set to 1, 2 or 4 (cyan, orange, and green, respectively). (A) shows the results for a low resolution ontology, shown above the plot. Each point is the average result for four randomly generated graphs with equivalent parameters. Grey lines indicate a power-law fit to the measurements. Dashed sections indicate extrapolation of the fit. (B) indicates the same measurement as (A) for a more descriptive ontology, shown above the plot. The inset of (B) shows the measurement again for an even more branching ontology, and indicates that the number of random matches decays catastrophically as the ontology size increases. Error bars are standard error of the mean.

For each graph and each ontology, pattern search was performed using the CEP algorithm. The CEP radius (number of interstitial nodes) was set to 1, 2, and 4 to explore the dependence of pattern detection on pattern flexibility. This created a total of 36 different search results for each ontology, with each result composed of the average of four trials (3 graph types $\times$ 4 sizes $\times$ 3 radii). For each graph type at each ontology resolution, the results were fit using a power-law distribution ($f(x) = \gamma x^r$). The results are shown in Figure 2. The average R^2 values of the fit were 0.80 (including the direct BA search, with low numbers of matches) or 0.95 (without the direct BA search results) for the minimal ontology, and 0.99 for the low-resolution ontology.

For graph and ontology combinations in which search results were found for $>$ 3 of the graph sizes, our results indicate that the number of pattern isomorphisms found in an algorithmically generated graph can be modeled by a power law. Further, different graph types exhibit different fit coefficients, suggesting that certain graph and ontology combinations may have characteristic power-law parameters. Note that in several simulated parameter sets, fitting could not be performed due to the lack of detected pattern matches in the search data; this was the case for heavily branching (i.e. high resolution) ontologies. An example of the pattern matches found for a high resolution ontology can be seen in the inset of Figure 2B. The lack of pattern matches indicates that for even relatively small ontologies, graphs must be very large or very widely searched (i.e. with many interstitial nodes) before events begin to match patterns by random chance. Future work will incorporate additional ontologies to refine the analysis.

Expected number of subgraphs in random models

Although the general question ”how likely is it that this pattern appears by chance?” is extremely challenging in directed, labeled tactical networks associated with arbitrary ontologies, we can gain some insight into this problem by examining some simpler cases. The prevalence of simple patterns (such as triangles) in network models has been studied before and the standard clustering coefficient of a network is one measure of this [18]. Triangles have a particularly straightforward interpretation in social networks: if nodes represent people and links represent friendship, then closed triangles represent the situation where two friends have a mutual friend in common. Open triangles indicate two friends who dont know each other, or know and antagonize each other. Comparing these calculations between different social networks, or social networks and other interaction networks such as citations or organization charts, could relate to the way that information is disseminated. It also applies to tactical networks, in which instead of friends, the elements are transmitters. An open triangle then might indicate a link with no redundancy, or a link under attack. And as we discuss elsewhere in this paper, considering patterns that are more complex than triangles has important ramifications for intelligence analysis. Here we review how to calculate the expected number of triangles in the ER model, and show how this can be extended to an arbitrary pattern.

It is helpful to introduce a few formal definitions at this point. A graph G has a set of nodes $\mathbb{N}$ and a set of links $\mathbb{E}$ with corresponding sizes N and E. Two graphs are the same if they have the same nodes and the same links. Suppose graph G has two nodes a and b with one link $a-b$, and graph H has two nodes α and β with one link $\alpha-\beta$. Because they do not have the same nodes, these two graphs are not equivalent, even though they do have the same structure. The process used to

generate the surrogate data in our empirical method is a form of renaming in which the names are drawn from a predetermined library of elements. If there is a naming f such that $f(G) = H$, then G and H are isomorphic. A naming such that $f(G) = G$ is an automorphism of G. A subgraph of G is any graph that has some of the nodes and some of the links from G. We care particularly about induced subgraphs, which contain some set of the nodes in G, $\mathbb{S}$, and all the links that connect any pair of nodes in $\mathbb{S}$. The subgraph induced by $\mathbb{S}$ is $G|\mathbb{S}$.

Number of triangles in Erdős–Rényi graphs The classic random graph model, Erdős–Rényi graphs, is formed from either one of two closely related probability distributions, $G(n,p)$ and $G(n,M)$. $G(n,p)$ is constructive: start with n labeled nodes and iterate through every pair of nodes; for each pair (i,j), add a link $i-j$ to the graph with equal probability p. $G(n,M)$ is declarative: consider the ensemble of all graphs on n labeled nodes with exactly M links; choose one graph from this ensemble at random. It is clear that these two models are not identical (two selections from $G(n,p)$ won't necessarily have the same number of links, for example), but in general they behave the same way as n. In the empirical analysis, we use the $G(n,p)$ form, but for probability analysis we use the $G(n,M)$ form.

We begin by picking (uniformly and at random) one graph from $G(n,M)$. In this graph, the probability that any particular i,j,k triple of nodes is a closed triangle is

$$\frac{\text{\# of graphs where i,j,k is a triangle}}{\text{\# total}} \tag{1}$$

How many graphs are in the ensemble? Since there are $\binom{n}{2}$ distinct pairs of nodes and M are connected, there are in total $\binom{\binom{n}{2}}{M}$ different graphs in the $G(n,M)$ ensemble. In how many of those is i,j,k a closed triangle? Imagine pulling all such cases out of the stack of graphs. In each of these selected graphs, three of the M links have been fixed, but the other links are unrestricted. There are $\binom{\binom{n}{2}-3}{M-3}$ ways to arrange the "un-fixed" links among the unused pairs of nodes.

This lets us write the probability that any particular triple is a closed triangle as

$$f_\Delta = \binom{\binom{n}{2}-3}{M-3} \Big/ \binom{\binom{n}{2}}{M} \tag{2}$$

Since there are $\binom{n}{3}$ distinct triples, the number of labeled triangular graphs we expect to see in a random graph of size n is

$$\binom{n}{3} \cdot f_\Delta = \frac{4}{3}\frac{M(M-1)(M-2)}{n^3-5n-4} \tag{3}$$

What about G(n,p)? Since each link is included at random independently of the others, the probability that any particular triple of nodes is a closed triangle is just $\binom{p}{3}$. This is an easier analysis, but is only easier because the graph itself is an easier model. For most real systems, they are not modeled well with Erdős–Rényi graphs, and the analysis of $G(n,M)$ is easier to extend to other graph models.

Arbitrary patterns Triangles form a useful example for studying, but we are more interested in arbitrary patterns of node size k. Suppose I draw a data graph from $G(n,M)$ and have an arbitrarily

specified pattern graph P which contains k nodes and l links. Pick some particular subset $\mathbb{S}$ of the nodes of G such that the size of $\mathbb{S}$ is k. The probability that $G|\mathbb{S}$ is isomorphic to P is, as in Eq. 1,

$$\frac{\text{\# of graphs where the graphs are isomorphic}}{\text{\# total}} \tag{4}$$

How many members of the ensemble exist such that $G|\mathbb{S} \cong P$? This question becomes easier if we introduce the concept of naming. A naming f is just a mapping or one-to-one function from one set of node labels to another. We write $f(G)$ to indicate the new graph that you get by renaming all the nodes and links in G according to f. Namings are invertible: $f^{-1} \cdot f(G) = G$.

If we pick some particular naming f from the nodes of P to the nodes in $\mathbb{S}$, then how many members of the ensemble exist such that $G|\mathbb{S} \cong P$? As before, a certain number of links and a certain number of node pairs are fixed, so the number of graphs satisfying this condition is equal to the number of ways to arrange the remaining links among the remaining node pairs, which is

$$\binom{\binom{n}{2} - \binom{k}{2}}{M - l} \tag{5}$$

where M is defined as before. This is the number of graphs in the ensemble that have $G|\mathbb{S}$ equal to P under one particular naming. There are $k!$ different namings, and if if we consider a different naming, say g instead of f, then it might point to a different subset of the ensemble, although this is not necessarily required. The total number of possible automorphic namings for a particular pattern P is the size of the automorphism group of the pattern graph, $Aut(P)$.

Computing the automorphism group $Aut(P)$ is currently thought to be as hard as the graph isomorphism problem. For real-world graphs, analytical results will not be easily derivable, and there are no particularly efficient algorithms for computing it. But, since we expect pattern graphs to be moderately sized compared to the data graphs, this may be more difficult from a theoretical perspective than from a practical one. For a small pattern graph, computing $Aut(P)$ will likely be feasible. The calculation of $Aut(P)$ leads us to the final result, which states that the probability that the induced subgraph for any particular set of nodes $\mathbb{S}$ is isomorphic to P is

$$f_p = \frac{k!}{|Aut(P)|} \cdot \binom{\binom{n}{2} - \binom{k}{2}}{M - l} \Bigg/ \binom{\binom{n}{2}}{M} \tag{6}$$

This is the likelihood that for a remapping or renaming, a pattern graph P would be remapped to an automorphism; in other words, it defines how likely a particular subgraph of size (k,l) is to appear by chance in a random graph of size n. This analysis is not yet general enough to map to the arbitrary pattern form used in our empirical analysis, but it provides a starting point for further theoretical understanding of the phenomena of labeled graph automorphisms being generated by random relabeling.

It is important to note that this analysis does not address the question of ontological resolution, and therefore cannot be directly applied to our empirical results. However, we believe that f_p is a useful metric for future analytical research into the behavior of pattern finding in labeled graphs. Future work in this area will include the influence of the ontology on the statistical prediction, as well as analyzing the influence of attribute on the likelihood of pattern matches. The difficult analytical nature of this problem underscores the need for an operationally useful empirical method.

Applications and operational relevance

In our empirical analysis, random graph structures were generated for both the pattern and the graph data, which were then imbued with ”meaning” by assigning categories drawn from an artificial ontology. Although this may not appear to have immediate operational relevance, the ability to make surrogate data has the potential to provide a valuable confidence measure in real-world situations. In a surrogate data set, the output graphs have the same number of nodes, links and edges, with the same distribution of category labels. However, the connectivity of the entities themselves are different each time the process is applied, as the node receiving a particular name and label will vary for each application of the naming function f. Essentially, the bulk statistics of the graph are maintained, but the fine structure of the graph is randomized. When this process is applied to a real-world situational description, the probability of a rare (and hence interesting) event appearing in the randomized graph is low (see previous section for an analytical description of a similar problem). Our method represents a first step toward computing an operationally feasible confidence bound on pattern matches found in labeled graphs.

To implement this method, a dataset (tactical, network, social, or other) and a search pattern are chosen. The pattern search function is a binary operation: either the event of interest described by the pattern is found or not. To determine the confidence of a match, many data surrogates are created with the same statistics; this could be performed empirically by reassignment of the original label set after shuffling. Each surrogate graph is then searched for the pattern of interest, and the number of subgraph isomorphisms counted. The appearance of a pattern match in the surrogate data would represent a match by chance, equivalent to a false positive. After the search is completed in each graph, the statistics of the number of subgraphs detected at each iteration can be used to compute a confidence interval, describing the expected number of statistically random pattern matches for the specific situation under analysis. We propose a definition of pattern match false positive likelihood as

$$p_f = \frac{\hat{n}_s}{\hat{n}_s + n_0} \tag{7}$$

where $\hat{n}_s$ is the average number of pattern matches in the surrogate data, and n_0 is the number of pattern matches in the original data. p_f is the likelihood that any particular pattern found in the original data is the result of random occurrences.

For a large set of surrogate data graphs, this empirical method may result in one of three outcomes: there may be the same number of average pattern matches in the surrogate data, substantially fewer, or substantially more. In the first case, $p_f = 0.5$, it is equally likely that a pattern match in the original graph is real or false; this indicates that the pattern match is likely a natural, random occurrence and is characteristic of the data set and ontology chosen. In cases for which the number of pattern matches in the surrogate data is substantially fewer and Equation 7 approaches zero, it indicates that the pattern match in the original data is likely genuine. If hundreds or thousands of surrogate graphs are explored, and a pattern match is never found, then it is unlikely that a detected match in the original data is a false positive. Finally, in cases for which there are substantially more pattern matches in the surrogate data, p_f will approach 1; this may be an indicator of active deception or sensor failure. If many matches are found in the surrogate data, then a reduced number of appearances in the measured data is unlikely to occur naturally. These three cases are shown in Figure 3. Each of these three possible outcomes provides guidance for additional action. The appearance of similar numbers of matching subgraphs in the surrogate data indicates that the

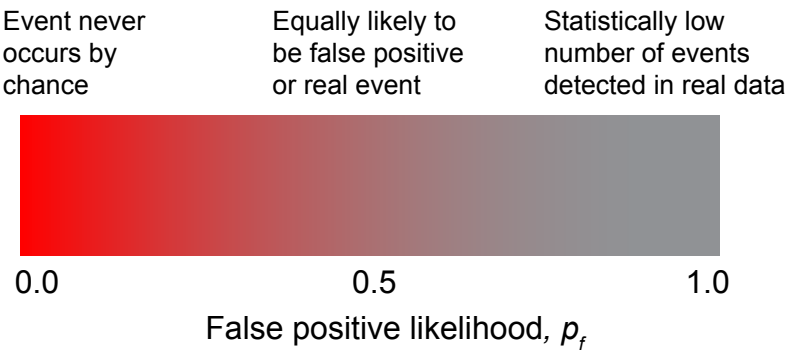

Figure 3. False positive likelihood.
p_f is computed from Eq. 7. The false positive likelihood will range from 0 to 1. A value of 0 indicates that an event detected in the original dataset is not detected in any of a large number of surrogate datasets. A value of 0.5 indicates an equal number of events identified in the original and each surrogate data is equal, meaning that a detected event is indistinguishable from a chance event. As p_f approaches 1, it indicates that an unusually low number of events was detected in the real data, and that there may be active interference with the measurement set.

event being searched for cannot be uniquely located in the data. This implies that the pattern must be made more specific, the ontology must be refined with additional categories, or that the data graph must be made larger by gathering of additional intelligence. When few pattern matches are found in the surrogate data, it provides justification with a quantifiable confidence for action based on the intelligence set. When many matches are found in the surrogate data, it may indicate the presence of a situation more complex than previously assumed or interference with the intelligence gathering. Crucially, the method presented here may provide guidance for the very difficult question of ”when is enough intelligence enough?” - a question that is not currently addressed with objective measurement techniques. By extrapolating the power-law fit of pattern matches vs graph size, it is possible to identify a range of graph sizes for which no pattern match would be expected to occur randomly. The center of this range is indicated by x-intercept of the solid lines in Figure 2. During operational use of this method, analysis of which canonical graph types best simulate real data could provide additional guidance on the amount of intelligence needed to observe an event while minimizing the chance of detecting a false positive. Conversely, the observation that more specific ontologies lead to fewer random pattern matches (see Figure 2) suggests that in an operational environment, the ability to scale up or down the specificity of an ontology would be useful - we could gain confidence about a pattern match in observed data by checking to see if the match is still there under a more specific ontology.

There are several important extensions of this preliminary research that warrant further study based on our results. First, it would be valuable to apply this method to a wider variety of graph sizes and ontology sizes. This would provide a stronger indication that the power-law models used are the best model for the data. This validation may eventually be provided by further mathematical analysis. Additionally, larger datasets based on ”real” (in the sense of generated by hu-

mans, whether during an exercise or an operation) are needed, especially those that contain some ground truth that is not readily visible by inspection. Although our analysis focused on a series of canonical graph types that are easily generated by application of an algorithm, a real-world situational description in graph form with thousands of nodes may not conform to one of these specific graph types. From an empirical standpoint, this point is not particularly important; any randomization of graph labels produces a valid surrogate. From a theoretical standpoint, it is possible that for unusual graph connectivities, the isomorphisms vs size curve may be non-monotonic. In such a situation, a power-law fit would no longer be valid, and the indicated false positive rates could be misleading. Further study of real-world datasets should reveal if such graphs appear within the typical scope of pattern-finding applications. Additionally, a complete analytical model of labeled graphs linked to ontologies could potentially result in more rapid algorithms for determining pattern match confidence.

Summary

The empirical analysis of pattern matches in surrogate data graphs provides a traceable confidence metric for environments in which graph-based pattern matching will be used as a decision aid. We demonstrated our method in the analysis of algorithmically-generated randomly labeled graphs, and have shown that the need for confidence metrics becomes critical as graph sizes increase. The metric presented in this analysis is clearly valuable in tactical situations, in which commanders may be asked to make rapid decisions based on large sets of intelligence data which are not intuitively accessible. It is also likely to be valuable for applications in counter-ISR, counter-terrorism, maritime objects of interest, business strategies, and wherever else multiple data types are fused into a graph and searched for patterns.

Confidence metrics also provide guidance for additional actions to be taken as the result of an event pattern detection. This could help to minimize risk to assets and personnel by preventing over-collection of data in the field. The use of our metric has the potential to reduce risk caused by decisions based on undeterminable questions, in which a chance occurrence of an event pattern is assumed to be a real event. We believe that implementation of confidence bounds will make graph-based search more useful, more appealing to end users, and contribute less risk to evidence-based decision making.

Acknowledgements

The authors gratefully acknowledge the support of SPAWAR Systems Center Pacific through the New Professional Program and Office of Naval Research.

References

[1] Richard J Aldrich. Policing the Past: Official History, Secrecy and British Intelligence since 1945.

[2] Albert-László Barabási and Réka and Albert. Emergence of Scaling in Random Networks. *Science (New York, NY)*, 286(5439):509–512, October 1999.

[3] A K Cebrowski and J J Garstka. Network-centric warfare: Its origin and future. *US Naval Institute Proceedings*, 1998.

[4] L P Cordella, P Foggia, C Sansone, and M Vento. A (sub)graph isomorphism algorithm for matching large graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(10):1367–1372, October 2004.

[5] Reinhard Diestel. *Graph Theory*. Springer, January 2000.

[6] P Erdős and A Rényi. On random graphs. *Publicationes Mathematicae Debrecen*, 6:290–297, 1959.

[7] Wenfei Fan, Jianzhong Li, Shuai Ma, Nan Tang, Yinghui Wu, and Yunpeng Wu. Graph pattern matching: from intractable to polynomial time. *Proceedings of the VLDB Endowment*, 3(1-2):264–275, September 2010.

[8] MJAG Langley and B Army. Network-Centric Warfare. *Military Review*, 2004.

[9] Jim Law and Scott McGirr. Multi-INT Complex Event Processing using Approximate, Incremental Graph Pattern Search. *17th Annual ICCRTS*, 2012.

[10] Ben Macintyre. *Operation Mincemeat*. Random House LLC, May 2010.

[11] James Moffat. *Complexity Theory and Network Centric Warfare*. DIANE Publishing, January 2010.

[12] Ewen Montagu. *The Man who Never was*. The Story of Operation Mincemeat. London : Evans, 1953.

[13] Thomas J Owens. Survey of Event Processing. *In-house technical memo.*, December 2007.

[14] D Popescu. Impact analysis for event-based components and systems. In *Software Engineering, 2010 ACM/IEEE 32nd International Conference on*, pages 401–404, 2010.

[15] Michael Smith. *The Secrets of Station X*. How the Bletchley Park Codebreakers Helped Win the War. Dialogue, 2011.

[16] Hanghang Tong, Christos Faloutsos, Brian Gallagher, and Tina Eliassi-Rad. Fast best-effort pattern matching in large attributed graphs. In *the 13th ACM SIGKDD international conference*, pages 737–746, New York, New York, USA, 2007. ACM Press.

[17] Segev Wasserkrug, Avigdor Gal, Opher Etzion, and Yulia Turchin. Complex event processing over uncertain data. In *the second international conference*, pages 253–264, New York, New York, USA, 2008. ACM Press.

[18] Duncan J Watts and Steven H Strogatz. Collective dynamics of —[lsquo]—small-world—[rsquo]— networks. *Nature*, 393(6684):440–442, June 1998.